PCT/EP2003/013394

Method for describing the composition of audio signals

The invention relates to a method and to an apparatus for coding and decoding a presentation description of audio signals, especially for the spatialization of MPEG-4 encoded audio signals in a 3D domain.

Background

The MPEG-4 Audio standard as defined in the MPEG-4 Audio standard ISO/IEC 14496-3:2001 and the MPEG-4 Systems standard 14496-1:2001 facilitates a wide variety of applications by supporting the representation of audio objects. For the combination of the audio objects additional information - the so-called scene description - determines the placement in space and time and is transmitted together with the coded audio objects.

For playback the audio objects are decoded separately and composed using the scene description in order to prepare a single soundtrack, which is then played to the listener.

For efficiency, the MPEG-4 Systems standard ISO/IEC 14496--1:2001 defines a way to encode the scene description in a binary representation, the so-called Binary Format for Scene Description (BIFS). Correspondingly, audio scenes are described using so-called AudioBIFS.

A scene description is structured hierarchically and can be represented as a graph, wherein leaf-nodes of the graph form the separate objects and the other nodes describes the processing, e.g. positioning, scaling, effects. The appearance and behavior of the separate objects can be controlled using parameters within the scene description nodes.

Invention

The invention is based on the recognition of the following fact. The above mentioned version of the MPEG-4 Audio standard defines a node named "Sound" which allows spatialization of audio signals in a 3D domain. A further node with the name "Sound2D" only allows spatialization on a 2D screen. The use of the "Sound" node in a 2D graphical player is not specified due to different implementations of the properties in a 2D and 3D player. However, from games, cinema and TV applications it is known, that it makes sense to provide the end user with a fully spatialized "3D-Sound" presentation, even if the video presentation is limited to a small flat screen in front. This is not possible with the defined "Sound" and "Sound2D" nodes.

Therefore, a problem to be solved by the invention is to overcome the above mentioned drawback. This problem is solved by the coding method disclosed in claim 1 and the corresponding decoding method disclosed in claim 5.

In principle, the inventive coding method comprises the generation of a parametric description of a sound source including information which allows spatialization in a 2D coordinate system. The parametric description of the sound source is linked with the audio signals of said sound source. An additional 1D value is added to said parametric description which allows in a 2D visual context a spatialization of said sound source in a 3D domain.

Separate sound sources may be coded as separate audio objects and the arrangement of the sound sources in a sound scene may be described by a scene description having first nodes corresponding to the separate audio objects and second nodes describing the presentation of the audio objects. A field of a second node may define the 3D spatialization of a

sound source.

Advantageously, the 2D coordinate system corresponds to the screen plane and the 1D value corresponds to a depth information perpendicular to said screen plane.

Furthermore, a transformation of said 2D coordinate system values to said 3 dimensional positions may enable the movement of a graphical object in the screen plane to be mapped to a movement of an audio object in the depth perpendicular to said screen plane.

The inventive decoding method comprises, in principle, the reception of an audio signal corresponding to a sound source linked with a parametric description of the sound source. The parametric description includes information which allows spatialization in a 2D coordinate system. An additional 1D value is separated from said parametric description. The sound source is spatialized in a 2D visual contexts in a 3D domain using said additional 1D value.

Audio objects representing separate sound sources may be separately decoded and a single soundtrack may be composed from the decoded audio objects using a scene description having first nodes corresponding to the separate audio objects and second nodes describing the processing of the audio objects. A field of a second node may define the 3D spatialization of a sound source.

Advantageously, the 2D coordinate system corresponds to the screen plane and said 1D value corresponds to a depth information perpendicular to said screen plane.

Furthermore, a transformation of said 2D coordinate system values to said 3 dimensional positions may enable the movement of a graphical object in the screen plane to be mapped

to a movement of an audio object in the depth perpendicular to said screen plane.

Exemplary embodiments

The Sound2D node is defined as followed:

```
Sound2D {
    exposedField    SFFloat    intensity    1.0
    exposedField    SFVec2f    location     0,0
    exposedField    SFNode     source       NULL
    field           SFBool     spatialize   TRUE
}
```

and the Sound node, which is a *3D* node, is defined as followed:

```
Sound {
    exposedField    SFVec3f    direction    0, 0, 1
    exposedField    SFFloat    intensity    1.0
    exposedField    SFVec3f    location     0, 0, 0
    exposedField    SFFloat    maxBack      10.0
    exposedField    SFFloat    maxFront     10.0
    exposedField    SFFloat    minBack      1.0
    exposedField    SFFloat    minFront     1.0
    exposedField    SFFloat    priority     0.0
    exposedField    SFNode     source       NULL
    field           SFBool     spatialize   TRUE
}
```

In the following the general term for all sound nodes (Sound2D, Sound and DirectiveSound) will be written in lower-case e.g. 'sound nodes'.

In the simplest case the Sound or Sound2D node is connected

via an AudioSource node to the decoder output. The sound nodes contain the *intensity* and the *location* information.

From the audio point of view a sound node is the final node before the loudspeaker mapping. In the case of several sound nodes, the output will be summed up. From the systems point of view the sound nodes can be seen as an entry point for the audio sub graph. A sound node can be grouped with non-audio nodes into a Transform node that will set its original location.

With the *phaseGroup* field of the AudioSource node, it is possible to mark channels that contain important phase relations, like in the case of "stereo pair", "multichannel" etc. A mixed operation of phase related channels and non-phase related channels is allowed. A *spatialize* field in the sound nodes specifies whether the sound shall be spatialized or not. This is only true for channels, which are not member of a phase group.

The Sound2D can spatialize the sound on the 2D screen. The standard said that the sound should be spatialized on scene of size 2m x 1.5m in a distance of one meter. This explanation seems to be ineffective because the value of the location field is not restricted and therefore the sound can also be positioned outside the screen size.

The Sound and DirectiveSound node can set the *location* everywhere in the 3D space. The mapping to the existing loudspeaker placement can be done using simple amplitude panning or more sophisticated techniques.

Both Sound and Sound2D can handle multichannel inputs and basically have the same functionalities, but the Sound2D node cannot *spatialize* a sound other than to the front.

A possibility is to add Sound and Sound2D to all scene graph profiles, i.e. add the Sound node to the SF2DNode group.

But, one reason for not including the "3D" sound nodes into the 2D scene graph profiles is, that a typical 2D player is not capable to handle 3D vectors (SFVec3f type), as it would be required for the Sound *direction* and *location* field.

Another reason is that the Sound node is specially designed for virtual reality scenes with moving listening points and attenuation attributes for far distance sound objects. For this the Listening point node and the Sound *maxBack*, *maxFront*, *minBack* and *minFront* fields are defined.

According one embodiment the old Sound2D node is extended or a new Sound2Ddepth node is defined. The Sound2Ddepth node could be similar the Sound2D node but with an additional *depth* field.

```
Sound2Ddepth {
    exposedField    SFFloat     intensity   1.0
    exposedField    SFVec2f     location    0,0
    exposedField    SFFloat     depth       0.0
    exposedField    SFNode      source      NULL
    field           SFBool      spatialize  TRUE
}
```

The *intensity* field adjusts the loudness of the sound. Its value ranges from 0.0 to 1.0, and this value specifies a factor that is used during the playback of the sound.

The *location* field specifies the location of the sound in the 2D scene.

The *depth* field specifies the depth of the sound in the 2D scene using the same coordinate system than the location

field. The default value is 0.0 and it refers to the screen position.

The *spatialize* field specifies whether the sound shall be spatialized. If this flag is set, the sound shall be spatialized with the maximum sophistication possible.

The same rules for multichannel audio spatialization apply to the Sound2Ddepth node as to the Sound (3D) node.

Using the Sound2D node in a 2D scene allows presenting surround sound, as the author recorded it. It is not possible to *spatialize* a sound other than to the front. Spatialize means moving the location of a monophonic signal due to user interactivities or scene updates.

With the Sound2Ddepth node it is possible to *spatialize* a sound also in the back, at the side or above of the listener. Supposing the audio presentation system has the capability to present it.

The invention is not restricted to the above embodiment where the additional *depth* field is introduced into the Sound2D node. Also, the additional *depth* field could be inserted into a node hierarchically arranged above the Sound2D node.

According to a further embodiment a mapping of the coordinates is performed. An additional field dimensionMapping in the Sound2DDepth node defines a transformation, e.g. as a 2 rows x 3 columns Vector used to map the 2D context coordinate-system (*ccs*) from the ancestor's transform hierarchy to the origin of the node.
The node's coordinate system (*ncs*) will be calculated as follows:

$$ncs = ccs \times dimensionMapping.$$

The location of the node is a 3 dimensional position, merged from the 2D input vector location and depth {location.x location.y depth} with regard to ncs.

Example: The node's coordinate system context is $\{x_i, y_i\}$. dimensionMapping is {1, 0, 0, 0, 0, 1}. This leads to ncs={ x_i, 0, y_i}, what enables the movement of an object in the y-dimension to be mapped to the audio movement in the depth.

The field 'dimensionMapping' may be defined as MFFloat. The same functionality could also be achieved by using the field data type 'SFRotation' that is an other MPEG-4 data type.

The invention allows the spatialization of the audio signal in a 3D domain, even if the playback device is restricted to 2D graphics.